

WHAT IS CLAIMED IS:

1. A method in a computer system for failover when at least one of a first network adapter and a data path through the first network adapter fails, wherein the computer system includes a filter driver, and wherein the first network adapter is connected to a second network adapter, comprising:

receiving, with the filter driver, a path fail notification that at least one of the first network adapter and the data path through the first network adapter has failed; and

rerouting, with the filter driver, packets directed to the first network adapter to the second network adapter.

2. The method of claim 1, further comprising:

changing, with the filter driver, a success status of each packet that had been directed to the first network adapter before the path fail notification was received to a busy status.

3. The method of claim 1, further comprising:

determining, with the filter driver, a new data path including the second network adapter.

4. The method of claim 1, further comprising:

receiving, with the filter driver, a notification that the first network adapter is restored; and

determining, with the filter driver, a data path for new data packets based on whether the notification specified a new data path.

5. The method of claim 1, further comprising:

designating, with the filter driver, a first storage device stack as a primary storage device stack in response to building the first storage device stack for a logical unit; and

designating, with the filter driver, a subsequent storage device stack as a secondary storage device stack in response to building the subsequent storage device stack for the logical unit.

6. The method of claim 5, further comprising:

preventing, with the filter driver, file system mounting on the secondary storage device stack.

7. A method for failover when at least one of a network adapter and a data path through the network adapter fails, wherein the network adapter is connected to a miniport driver that is connected to a filter driver, comprising:

determining, with the miniport driver, that the network adapter has failed; and
notifying, with the miniport driver, the filter driver that the network adapter has failed.

8. The method of claim 7, further comprising:

specifying, with the miniport driver, a new data path to be used by the filter driver for rerouting packets.

9. The method of claim 7, further comprising:

completing, with the miniport driver, processing of pending packets directed to the first network adapter with a success status.

10. The method of claim 7, further comprising:

determining, with the miniport driver, that the network adapter is restored; and
notifying, with the miniport driver, the filter driver that the network adapter is restored.

11. A method for static load balancing implemented in a filter driver, wherein the filter driver performs:

determining a data quota for each of multiple data paths;
identifying a maximum number of commands for a target logical unit; and
selecting a data path on which to send a packet based on the data quota and the maximum number of commands.

12. The method of claim 11, wherein the filter driver further performs:

determining a load balancing share associated with each data path in a team; and
determining a mean data length of transmitted packets for each data path.

13. The method of claim 11, wherein the data path is selected using a round robin technique, and wherein packets are sent along the selected data path until at least

one of the data quota for the data path is met and until a maximum number of commands is reached.

14. A method for dynamic load balancing implemented in a filter driver, wherein the filter driver performs:

determining a data transfer speed for each of multiple data paths;

updating a load balancing share for each of the multiple data paths based on the data transfer speed of each of the multiple data paths; and

selecting a data path on which to send a packet based on the load balancing share of each of the multiple data paths.

15. The method of claim 14, wherein the data transfer speed is calculated for each of the multiple data paths as a ratio of total data transferred and total time spent on transferring data.

16. The method of claim 14, wherein a frequency at which the data transfer speed is determined is tunable.

17. A system coupled to a network and data storage, comprising:

a host computer;

a storage controller managing Input/Output (I/O) access to the data storage, wherein the storage controller is coupled to the host computer;

a first network adapter

a second network adapter;

and

a filter driver at the host computer, wherein the filter driver is capable of: receiving a path fail notification that at least one of the first network adapter and a data path through the first network adapter has failed and rerouting packets directed to the first network adapter to the second network adapter.

18. The system of claim 17, wherein the filter driver is further capable of: changing a success status of each packet that had been directed to the first network adapter before the path fail notification was received to a busy status.

19. The system of claim 17, wherein the filter driver is further capable of:

determining a new data path including the second network adapter.

20. The system of claim 17, wherein the filter driver is further capable of:
receiving a notification that the first network adapter is restored; and
determining a data path for new data packets based on whether the notification
specified a new data path.

21. The system of claim 17, wherein the filter driver is further capable of:
designating a first storage device stack as a primary storage device stack in
response to building the first storage device stack for a logical unit; and
designating a subsequent storage device stack as a secondary storage device
stack in response to building the subsequent storage device stack for the logical unit.

22. The system of claim 21, wherein the filter driver is further capable of:
preventing file system mounting on the secondary storage device stack.

23. A system coupled to a network and data storage, comprising:
a host computer;
a storage controller managing Input/Output (I/O) access to the data storage,
wherein the storage controller is coupled to the host computer;
a filter driver at the host computer;
at least two network adapters at the host computer; and
a miniport driver at the host computer, wherein the miniport driver is capable of
determining that at least one of the network adapters has failed and notifying the filter
driver that the network adapter has failed.

24. The system of claim 23, wherein the miniport driver is further capable
of:

specifying a new data path to be used by the filter driver for rerouting packets.

25. The system of claim 23, wherein the miniport driver is further capable
of:

completing processing of pending packets directed to the failed network adapter
with a success status.

26. The system of claim 23, wherein the miniport driver is further capable of:

determining that the failed network adapter is restored; and
notifying the filter driver that the failed network adapter is restored.

27. A system coupled to a network and data storage, comprising:
a host computer;
a storage controller managing Input/Output (I/O) access to the data storage, wherein the storage controller is coupled to the host computer; and
a filter driver capable of determining a data quota for each of multiple data paths, identifying a maximum number of commands for a target logical unit, and selecting a data path on which to send a packet based on the data quota and the maximum number of commands.

28. The system of claim 27, wherein the filter driver further performs:
determining a load balancing share associated with each data path in a team; and
determining a mean data length of transmitted packets for each data path.

29. The system of claim 27, wherein the data path is selected using a round robin technique, and wherein packets are sent along the selected data path until at least one of the data quota for the data path is met and until a maximum number of commands is reached.

30. A system coupled to a network and data storage, comprising:
a host computer;
a storage controller managing Input/Output (I/O) access to the data storage, wherein the storage controller is coupled to the host computer; and
a filter driver capable of determining a data transfer speed for each of multiple data paths, updating a load balancing share for each of the multiple data paths based on the data transfer speed of each of the multiple data paths, and selecting a data path on which to send a packet based on the load balancing share of each of the multiple data paths.

31. The system of claim 30, wherein the data transfer speed is calculated for each of the multiple data paths as a ratio of total data transferred and total time spent on transferring data.

32. The system of claim 30, wherein a frequency at which the data transfer speed is determined is tunable.

33. An article of manufacture comprising a storage medium having stored therein instructions that when executed by a computing device results in the following:
receiving a path fail notification that at least one of a first network adapter and a data path through the first network adapter has failed; and

rerouting packets directed to the first network adapter to the second network adapter.

34. The article of manufacture of claim 33, wherein the instructions when executed further result in the following:

changing a success status of each packet that had been directed to the first network adapter before the path fail notification was received to a busy status.

35. The article of manufacture of claim 33, the instructions when executed further result in the following:

determining a new data path including the second network adapter.

36. The article of manufacture of claim 33, the instructions when executed further result in the following:

receiving a notification that the first network adapter is restored; and

determining a data path for new data packets based on whether the notification specified a new data path.

37. The article of manufacture of claim 33, the instructions when executed further result in the following:

designating a first storage device stack as a primary storage device stack in response to building the first storage device stack for a logical unit; and

designating a subsequent storage device stack as a secondary storage device stack in response to building the subsequent storage device stack for the logical unit.

38. The article of manufacture of claim 37, the instructions when executed further result in the following:

preventing file system mounting on the secondary storage device stack.

39. An article of manufacture comprising a storage medium having stored therein instructions that when executed by a computing device results in the following:
determining that the network adapter has failed; and
notifying the filter driver that the network adapter has failed.

40. The article of manufacture of claim 39, the instructions when executed further result in the following:

specifying a new data path to be used by the filter driver for rerouting packets.

41. The article of manufacture of claim 39, the instructions when executed further result in the following:

completing processing of pending packets directed to the first network adapter with a success status.

42. The article of manufacture of claim 39, the instructions when executed further result in the following:

determining that the network adapter is restored; and
notifying the filter driver that the network adapter is restored.

43. An article of manufacture comprising a storage medium having stored therein instructions that when executed by a computing device results in the following:

determining a data quota for each of multiple data paths;
identifying a maximum number of commands for a target logical unit; and
selecting a data path on which to send a packet based on the data quota and the maximum number of commands.

44. The article of manufacture of claim 43, the instructions when executed further result in the following:

determining a load balancing share associated with each data path in a team; and
determining a mean data length of transmitted packets for each data path.

45. The article of manufacture of claim 43, wherein the data path is selected using a round robin technique, and wherein packets are sent along the selected data path until at least one of the data quota for the data path is met and until a maximum number of commands is reached.

46. An article of manufacture comprising a storage medium having stored therein instructions that when executed by a computing device results in the following:

- determining a data transfer speed for each of multiple data paths;
- updating a load balancing share for each of the multiple data paths based on the data transfer speed of each of the multiple data paths; and
- selecting a data path on which to send a packet based on the load balancing share of each of the multiple data paths.

47. The article of manufacture of claim 46, wherein the data transfer speed is calculated for each of the multiple data paths as a ratio of total data transferred and total time spent on transferring data.

48. The system of claim 46, wherein a frequency at which the data transfer speed is determined is tunable